

Parareal algorithm via Chebyshev-Gauss spectral collocation method

Quan Zhou*

Yicheng Liu[†]

Shu-Lin Wu[‡]

Abstract. We present the Parareal-CG algorithm for time-dependent differential equations in this work. The algorithm is a parallel in time iteration algorithm utilizes Chebyshev-Gauss spectral collocation method for fine propagator \mathcal{F} and backward Euler method for coarse propagator \mathcal{G} . As far as we know, this is the first time that the spectral method used as the \mathcal{F} propagator of the parareal algorithm. By constructing the stable function of the Chebyshev-Gauss spectral collocation method for the symmetric positive definite (SPD) problem, we find out that the Parareal-CG algorithm and the Parareal-TR algorithm, whose \mathcal{F} propagator is chosen to be a trapezoidal ruler, converge similarly, i.e., the Parareal-CG algorithm converge as fast as Parareal-Euler algorithm with sufficient Chebyshev-Gauss points in every coarse grid. Numerical examples including ordinary differential equations and time-dependent partial differential equations are given to illustrate the high efficiency and accuracy of the proposed algorithm.

Key words. Parareal algorithm, Chebyshev-Gauss spectral collocation method, nonlinear ODEs, SPD problems, time-dependent PDEs.

MSC-classification. 65L05, 65L20, 65L60, 68Q60.

1 Introduction

We are considering utilizing the parareal algorithm for the initial-value problems in the following

$$\begin{cases} \frac{du}{dt} = f(t, u), & t \in [0, T], \\ u(0) = u_0. \end{cases} \quad (1.1)$$

where $f : (0, T) \times \mathbb{R}^m \rightarrow \mathbb{R}^m$, and $u_0 \in \mathbb{R}^m$. Parareal is a well-studied parallel in time algorithm developed by Lions, Maday, and Turinici in 2001 [27]. The algorithm obtains the solution in a limited number of predictor-corrector iterations utilizing random initial values at each temporal subinterval, stopping when a tolerance is reached. The global error produced by this iterative method is equivalent to that obtained by the serial

*College of Science, National University of Defense Technology, Changsha, Hunan 410073, China. quanzhoujune@163.com. Zhou is supported by the the Postgraduate Scientific Research Innovation Project of Hunan Province (No.CX20210012).

[†]College of Science, National University of Defense Technology, Changsha, Hunan 410073, China. liuyc2001@hotmail.com.

[‡]Corresponding author, School of Mathematics and Statistics, Northeast Normal University, Changchun 130024, China. wushulin84@hotmail.com.

of the fine propagator. Due to the advantages of parareal algorithm including but not limited to efficiency and convergence, many relevant methods have emerged in recent years [14, 27, 31, 43]. Together with these methods, the parareal algorithm has been identified in various fields of research, including optimal control problems [13, 30, 36, 38], wave equations [11, 20], stochastic differential equations [5, 22, 46], Hamiltonian systems [9, 15], incompressible flows [8, 10], heat equations [23, 25, 40], algebraic equations [16, 24], molecular dynamics [2, 26], and other partial differential equations [4, 28, 29].

The parareal algorithm combines two numerical methods which are the coarse propagator \mathcal{G} and fine propagator \mathcal{F} , associated with the large time step ΔT and small time step Δt , respectively. The ratio $J = \Delta T / \Delta t$ is assumed to be greater than 1. The \mathcal{G} propagator is often chosen to be backward Euler method which is inexpensive and strongly stable so that it is available for the large time step ΔT computations. Additionally, numerical methods based on Taylor's expansion and quadrature formula which is much cheaper are frequently chosen as \mathcal{F} propagators, and the convergence of various \mathcal{F} propagators has been thoroughly investigated for the symmetric positive definite (SPD) problem.

$$u'(t) + Au(t) = g(t), \quad A \in \mathbb{R}^{m \times m} \text{ symmetric positive definite (SPD)}. \quad (1.2)$$

Gander and Vandewalle [17] proposed the convergence theorem of parareal algorithm and illustrated numerically that Parareal-Euler algorithm using for \mathcal{F} the backward-Euler method converges rapidly. Based on which, Mathew, Sarkis and Schaerer [30] proved theoretically that the parareal-Euler algorithm converged robustly and the convergence factor is 0.298 for $J \geq 2$. Wu [35] demonstrated that the robust convergence also holds for \mathcal{F} propagator chosen to be second-order diagonal implicit Runge-Kutta (DIRK2) method and TR/BDF2 method (ode23tb solver for ODEs in Matlab). For $J \geq 2$, the convergence factor of the Parareal-2sDIRK and Parareal-TR/BDF2 are 0.316 and 0.333 respectively. Wu and Zhou [42] also showed the analysis \mathcal{F} propagator for the third-order diagonal implicit Runge-Kutta method with a convergence factor 0.333 for $J \geq 4$; they also proved that for trapezoidal formula and fourth-order Gauss-Runge-Kutta integrator chosen to be \mathcal{F} propagator, there exists a J_{\min}^* depends on both spectral radius of A and the step size Δt which make the convergence factor be 0.333 for $J \geq J_{\min}^*$. Recently, Yang, Yuan and Zhou [45] gave a more general result stating that if the \mathcal{F} propagator is strongly stable single step integrators, there must exist a positive J_{\min}^* (independent of step sizes ΔT , Δt , terminal time T , problem data u_0 and f , as well as the spectral radius of A) such that the parareal algorithm converges linearly with convergence factor close to 0.3 for all $J \geq J_{\min}^*$.

In this work, we choose Chebyshev-Gauss spectral collocation method [44] for fine propagator and present Parareal-CG algorithm. The Chebyshev-Gauss spectral collocation method is an overall iteration method with M Chebyshev-Gauss points in the computation interval, so it combines the advantages of spectral accuracy and computational efficiency. Also, it is unnecessary to solve implicit equations for every fine time step Δt as classical methods via Taylor's expansion mentioned in [17, 35, 42, 45]. We would briefly introduce the spectral collocation methods for ODEs. Clenshaw and Norton first presented the Chebyshev-Picard method for solving nonlinear ordinary differential equations in 1963, in which the nonlinear term was approximated by Chebyshev series, so that the method was collocated at Chebyshev-Gauss-Lobatto points and implemented by Picard iteration. Later, a matrix-vector form of the method was introduced by Feagin

and Nacozy [12], greatly increases the computation efficiency. Yang and Wang [44] proposed Chebyshev-Gauss spectral collocation method via Chebyshev-Gauss points for ODEs in a single interval and analyzed the convergence by the hp version. The method demonstrated significant advantages in astrodynamics simulations [1, 3, 37, 41] because of its spectral accuracy and computational efficiency. Additionally, some relevant spectral collocation methods for solving ODEs are also proposed in recent years [18, 19, 39].

We present the matrix-vector form of Chebyshev-Gauss spectral collocation method by the approach in [12]. Based on which we construct its stability function and illustrate the convergence of the Parareal-CG method numerically. We find out there exists a M_{\min}^* depends on spectral radius of A and the step size ΔT which makes the convergence factor be 0.333 for $M \geq M_{\min}^*$. The proposed method has the same convergence as Parareal-TR and Parareal-Gauss4 methods.

The rest of the paper is organized as follows, in section 2 we recall the Chebyshev-Gauss spectral collocation method and its convergence theorem. Parareal-CG algorithm is proposed in 3. We provide stability function of Chebyshev-Gauss spectral collocation method and observe the convergence of Parareal-CG algorithm in 4. Several numerical experiments are carried out in Section 5 to demonstrate the high accuracy and convergency of the proposed method. We finally give some conclusions in Section 6.

2 Revisit of Chebyshev-Gauss Spectral Collocation Method

In this section, we revisit the Chebyshev-Gauss spectral collocation method proposed by Yang and Wang [44] in a general interval $[a, b]$, ($b > a > 0$) with the initial condition $u(a) = u_a$.

Let $T_l(\tau) = \cos(l \arccos(\tau))$ be the standard Chebyshev polynomials of degree l , ($l = 0, 1, \dots$) with $\tau \in [-1, 1]$, then by using the affine transformation we can define the shifted Chebyshev polynomials

$$\tilde{T}_l(t) = T_l\left(\frac{2(t-a)}{b-a} - 1\right), \quad t \in [a, b], \quad l = 0, 1, \dots \quad (2.1)$$

According to the definition, one can obtain the following shifted Chebyshev derivative relationship directly

$$\begin{aligned} \tilde{T}'_0(t) &= 0, \quad \tilde{T}'_1(t) = \frac{2}{b-a}, \\ \frac{1}{l+1} \tilde{T}'_{l+1}(t) - \frac{1}{l-1} \tilde{T}'_{l-1}(t) &= \frac{4}{b-a} \tilde{T}_l(t), \quad l \geq 2. \end{aligned} \quad (2.2)$$

Let τ_m denote the standard Chebyshev-Gauss (CG) points in $(-1, 1)$,

$$\tau_m = -\cos \frac{(2m+1)\pi}{2M+2}, \quad m = 0, 1, \dots, M, \quad (2.3)$$

the corresponding shifted Chebyshev-Gauss (CG) points t_m has the form

$$t_m = \frac{b-a}{2} \tau_m + \frac{a+b}{2}, \quad m = 0, 1, \dots, M, \quad (2.4)$$

which are the zeros of $\tilde{T}_{M+1}(t)$.

Denote $\mathcal{P}_{M+1}(a, b)$ be a set of polynomials of degree at most $M+1$ in (a, b) , the Chebyshev-Gauss

spectral collocation method is to seek $u_M(t) \in \mathcal{P}_{M+1}(a, b)$ defined by

$$u_M(t) = \sum_{m=0}^{M+1} \hat{u}_m \tilde{T}_m(t), \quad (2.5)$$

such that

$$\begin{cases} \frac{d}{dt} u_M(t_m) = \mathcal{I}_M f(t_m, u_M(t_m)), & m = 0, 1, \dots, M, \\ u_M(a) = u_a, \end{cases} \quad (2.6)$$

where $\mathcal{I}_M f(t, u(t)) : C(a, b) \rightarrow \mathcal{P}_M(a, b)$ is the Chebyshev interpolation of $f(t, u(t))$ defined by

$$\mathcal{I}_M f(t, u_M(t)) = \sum_{m=0}^M \hat{f}_m \tilde{T}_m(t), \quad (2.7)$$

the coefficients $\{\hat{f}_m\}_{m=0}^M$ are determined by the forward discrete Chebyshev transform

$$\hat{f}_m = \frac{2}{c_m(M+1)} \sum_{l=0}^M f(t_l, u_M(t_l)) \tilde{T}_m(t_l), \quad (2.8)$$

where $c_0 = 2$, $c_m = 1$ for $m \geq 1$. Then by the shifted Chebyshev derivative relationship (2.2), we can derive the coefficients $\{\hat{u}_m\}_{m=0}^{M+1}$ in (2.5) by

$$\begin{aligned} \hat{u}_{M+1} &= \frac{(b-a)}{4(M+1)} \hat{f}_M, \quad \hat{u}_M = \frac{(b-a)}{4M} \hat{f}_{M-1}, \\ \hat{u}_m &= \frac{b-a}{4m} (c_{m-1} \hat{f}_{m-1} - \hat{f}_{m+1}), \quad 1 \leq m \leq M-1, \\ \hat{u}_0 &= u_a - \sum_{k=1}^{M+1} (-1)^k \hat{u}_k. \end{aligned} \quad (2.9)$$

Yang and Wang [44] presented the error estimate for the hp -version of the single interval Chebyshev-Gauss spectral collocation method, before introduce the theorem, we first present some notations used throughout the error estimate

- $H_\omega^r(a, b)$, ($r \geq 0$) denotes the weighted Sobolev space with certain weight function $\omega = (t-a)(b-t)$ in (a, b) , especially, $H_\omega^0(a, b) = L_\omega^2(a, b)$.
- $\|v\|_\omega$ denotes the norm of the space $L_{\omega^{-1/2}}^2(a, b)$.

The error estimate theorem is stated as follows.

Theorem 2.1 Assume that $f(t, z)$ fulfills the Lipschitz condition,

$$|f(z_1, t) - f(z_2, t)| \leq L|z_1 - z_2|, \quad L > 0, \quad (2.10)$$

and $0 < L(b-a) < \beta < \frac{1}{4}$ (β is a certain constant) holds. Then for any $u \in H_{\omega^{r-\frac{3}{2}}}^r(a, b)$ with integers

$2 \leq r \leq N + 1$, we have

$$\begin{aligned} \|u - u_M\|_{L^2(a,b)}^2 &\leq \frac{b-a}{2} \|u - u_M\|_{\omega}^2 \leq C_{\beta} (b-a)^3 M^{4-2r} \int_a^b \omega^{r-\frac{3}{2}}(t) \left(\frac{d^r}{dt^r} u(t) \right)^2 dt, \\ |u(b) - u_M(b)| &\leq C_{\beta} (b-a)^2 M^{4-2r} \int_a^b \omega^{r-\frac{3}{2}}(t) \left(\frac{d^r}{dt^r} u(t) \right)^2 dt. \end{aligned} \quad (2.11)$$

where C_{β} is a positive constant depending only on β .

In actual computation, we use Picard iteration procedure to compute the coefficients $\{\hat{u}_m\}_{m=0}^M$. The approach is simple to implement, especially for the complex nonlinear problems. The p -th ($p = 1, 2, \dots$) Picard iteration form of (2.6) is

$$\frac{d}{dt} u_M^{p+1}(t_m) = \mathcal{I}_M f(t_m, u_M^p(t_m)), \quad (2.12)$$

if the equation satisfies the convergence condition in Theorem 2.1, the iteration solution $u_M^p(t)$ will converge to the numerical solution $u_M(t)$ with large enough p , and the convergence is of order one, that is to say there exists a constant $0 < C_p < 1$ such that $\|e_{p+1}\|_{\infty} \leq C_p \|e_p\|_{\infty}$ with the definition $e_p := u_M^p(t) - u_M(t)$.

Algorithm 1 Chebyshev-Gauss Spectral Collocation Algorithm

Input: Provide the initial guess of $\{u_0^M(t_m)\}_{m=0}^M$, the tolerance ϵ .

For $p = 0, 1, \dots$

Step 1. Evaluate the values of $\{f(t_m, u_M^p(t_m))\}_{m=0}^M$,

Step 2. Compute the coefficients $\{\hat{f}_m^p\}_{m=0}^M$ by (2.8).

Step 3. Compute the coefficients $\{\hat{u}_m^p\}_{m=0}^{M+1}$ by (2.9).

Step 4. Update the data of $\{u_M^{p+1}(t_m)\}_{m=0}^M$ by (2.5).

Step 5. If the iteration error satisfies the stopping criterion,

$$\|u_M^{p+1}(t) - u_M^p(t)\|_{\infty} < \epsilon,$$

terminate the iteration; otherwise go back to Step 1.

Step 6. Compute $u_M^{p+1}(b) = \sum_{m=0}^{M+1} \hat{u}_m^p$.

We designate \mathcal{F}_{CG} as the numerical propagator defined by the Chebyshev-Gauss spectral collocation method at the conclusion of the section. The numerical output of Algorithm 1 with M points in the interval $[a, b]$ is represented as $\mathcal{F}_{CG}(a, u_a, M, b - a)$. That is,

$$u_M^{p+1}(b) = \mathcal{F}_{CG}(a, u_a, M, b - a). \quad (2.13)$$

3 Parareal-CG algorithm

The parareal algorithm introduced by Gander and Vandewalle [17] is revisited in this section, using the Chebyshev-Gauss spectral collocation method as the fine propagator. First, we divide the whole time interval $[0, T]$ uniformly by $0 = T_0 < T_1 < \dots < T_N = T$ and define $\Delta T = T/N$. Second, we divide each (T_n, T_{n+1}) by $M (\geq 2)$ shifted Chebyshev-Gauss points (2.4). Then, the low-order and inexpensive numerical method \mathcal{G} propagator is applied to the coarse time grids, while the Chebyshev-Gauss spectral collocation method \mathcal{F}_{CG}

propagator having spectral accuracy is utilized in the fine grids. The time-sequential and time-parallel parts of the algorithm are denoted by the symbols \ominus and \oplus , respectively. The following is the parareal method employing \mathcal{F}_{CG} as the fine propagator.

Algorithm 2 Parareal-CG Algorithm

- \ominus **Initialization:** Compute sequentially $u_{n+1}^0 = \mathcal{G}(T_n, u_n^0, \Delta T)$ with $u_0^0 = u_0$, $n = 0, 1, \dots, N-1$;
- For** $k = 0, 1, \dots$
- \oplus **Step 1.** On each subinterval $[T_n, T_{n+1}]$, compute $\tilde{u}_{n+1} = \mathcal{F}_{CG}(T_n, u_n^k, M, \Delta T)$.
- \ominus **Step 2.** Perform sequential corrections

$$u_{n+1}^{k+1} = \mathcal{G}(T_n, u_n^{k+1}, \Delta T) + \tilde{u}_{n+1} - \mathcal{G}(T_n, u_n^k, \Delta T), \quad (3.1)$$

where $u_0^{k+1} = u_0$, $n = 0, 1, \dots, N-1$;

- \ominus **Step 3.** If $\{u_n^{k+1}\}_{n=1}^N$ satisfies the stopping criterion, terminate the iteration and output $\{u_n^{k+1}\}_{n=1}^N$; otherwise go back to Step 1.
-

Given that the implicit Euler method is L -stable and feasible for high coarse time step sizes ΔT forced by the parareal algorithm, using it as the coarse propagator makes sense. Other reliable implicit numerical methods, such implicit Runge-Kutta methods, are also available for use as the \mathcal{G} propagator, although they are all significantly more costly than the implicit Euler method. In this paper, we apply the Chebyshev-Gauss spectral collocation method to the fine propagator, and the compact form of the Parareal-CG method may be expressed as

$$u_{n+1}^{k+1} = \mathcal{G}(T_n, u_n^{k+1}, \Delta T) + \mathcal{F}_{CG}(T_n, u_n^k, M, \Delta T) - \mathcal{G}(T_n, u_n^k, \Delta T). \quad (3.2)$$

For comparison, we also take into account other high-order and expensive numerical methods like trapezoidal formula. Accordingly, each interval $[T_n, T_{n+1}]$ should be divided into J (≥ 2) small time-intervals $[T_{n+\frac{j}{J}}, T_{n+\frac{j+1}{J}}]$, $j = 0, 1, \dots, J-1$. We assume the intervals are of uniform size, therefore, $\Delta t = \frac{\Delta T}{J}$. After that, the parareal algorithm can be derived by

Algorithm 3 Parareal Algorithm

- \ominus **Initialization:** Compute sequentially $u_{n+1}^0 = \mathcal{G}(T_n, u_n^0, \Delta T)$ with $u_0^0 = u_0$, $n = 0, 1, \dots, N-1$;
- For** $k = 0, 1, \dots$
- \oplus **Step 1.** On each subinterval $[T_n, T_{n+1}]$, compute $\tilde{u}_{n+\frac{j+1}{J}} = \mathcal{F}(T_{n+\frac{j}{J}}, \tilde{u}_{n+\frac{j}{J}}, \frac{\Delta T}{J})$ with initial value $\tilde{u}_n = u_n^k$, where $T_{n+\frac{j}{J}} = T_n + \frac{j\Delta T}{J}$ and $j = 0, 1, \dots, J-1$;
- \ominus **Step 2.** Perform sequential corrections

$$u_{n+1}^{k+1} = \mathcal{G}(T_n, u_n^{k+1}, \Delta T) + \tilde{u}_{n+1} - \mathcal{G}(T_n, u_n^k, \Delta T), \quad (3.3)$$

where $u_0^{k+1} = u_0$, $n = 0, 1, \dots, N-1$;

- \ominus **Step 3.** If $\{u_n^{k+1}\}_{n=1}^N$ satisfies the stopping criterion, terminate the iteration and output $\{u_n^{k+1}\}_{n=1}^N$; otherwise go back to Step 1.
-

The compact form of the parareal algorithm can be derived by

$$u_{n+1}^{k+1} = \mathcal{G}(T_n, u_n^{k+1}, \Delta T) + \mathcal{F}^J(T_n, u_n^k, \Delta t) - \mathcal{G}(T_n, u_n^k, \Delta T), \quad (3.4)$$

where $\mathcal{F}^J(T_n, u_n^k, \Delta t)$ stands for the result of running J steps of the fine propagator \mathcal{F} with initial value u_n^k and the small step-size Δt .

4 Convergence Analysis

Based on the parareal convergence analysis given by Gander and Vandewalle [17], the convergence of Parareal-CG method is analysed by constructing the stability function of Chebyshev-Gauss spectral collocation method in this section.

4.1 Stability function of Chevyshev-Gauss Spectral method

In order to obtain the convergence factor $\mathcal{K}(z, M)$ of Parareal-CG algorithm in Algorithm 2, in this subsection we present the stability function of Chebyshev-Gauss spectral collocation method $\mathcal{R}_{CG}(z, M)$.

Lemma 4.1 *For given M and $z := \lambda \Delta T$, the stability function $\mathcal{R}_{CG}(z, M)$ of Chebyshev-Gauss spectral collocation method in $[T_n, T_{n+1}]$, ($n = 1, 2, \dots, N$) is*

$$\mathcal{R}_{CG}(z, M) = \mathbf{T}(\mathbf{I}_2 - z\mathbf{C}_\alpha(\mathbf{I}_1 + z\mathbf{T}_1\mathbf{C}_\alpha)^{-1}\mathbf{T}_1)\mathbf{E}, \quad (4.1)$$

where \mathbf{T}_2 and \mathbf{C}_α are the coefficient matrices defined in (4.7) and (4.9), respectively; \mathbf{I}_1 and \mathbf{I}_2 are two identity matrices of the size $(M+1)$ and $(M+2)$, respectively; $\mathbf{T} = [1, 1, \dots, 1]_{1 \times (M+2)}$ and $\mathbf{E} = [1, 0, \dots, 0]_{(M+2) \times 1}^\top$ are two constant vectors.

Proof. We first introduce the matrix-vector form of the Chebyshev-Gauss spectral collocation method to be able to express \mathcal{R}_{CG} explicitly. Denote the evaluations of the approximated polynomial for the p -th ($p = 1, 2, \dots$) iteration $u_M^p(t)$ at the shifted Chebyshev-Gauss points $\{t_m\}_{m=0}^M$, ($n = 0, 1, \dots, N-1$) by

$$\mathbf{u}_p = [u_p^M(t_0), u_p^M(t_1), \dots, u_p^M(t_M)]_{(M+1) \times 1}^\top, \quad (4.2)$$

Equation (2.9) indicates that the vector \mathbf{u}_p can be expressed as a consequence of the fact $\mathcal{T}_l(t_m) = T_l(\tau_m)$, ($l = 1, 2, \dots; m = 0, 1, \dots, M$)

$$\mathbf{u}_p = \mathbf{T}_1 \hat{\mathbf{u}}_p, \quad (4.3)$$

where \mathbf{T}_1 is a coefficient matrix defined by

$$\mathbf{T}_1 = \begin{matrix} & (M+1) \times (M+2) \\ \begin{bmatrix} T_0(\tau_0) & T_1(\tau_0) & \cdots & T_{M+1}(\tau_0) \\ T_0(\tau_1) & T_1(\tau_1) & \cdots & T_{M+1}(\tau_1) \\ \vdots & \vdots & \ddots & \vdots \\ T_0(\tau_M) & T_1(\tau_M) & \cdots & T_{M+1}(\tau_M) \end{bmatrix} \end{matrix}. \quad (4.4)$$

Denote $\hat{f}(t_m, u_M^p(t_m)) = \hat{f}_m^p$ and suppose the initial value be $u(T_n) = u_{T_n}$, then the coefficient vector $\hat{\mathbf{u}}^p$ can be expressed by

$$\begin{aligned}
\hat{\mathbf{u}}^p &= [\hat{u}_0^p, \hat{u}_1^p, \dots, \hat{u}_{M+1}^p]_{(M+2) \times 1}^\top \\
&= \left[u_{T_n} + \sum_{m=1}^{M+1} (-1)^{m-1} \hat{u}_m^p, \frac{\Delta T}{4} (2\hat{f}_0^{p-1} - \hat{f}_2^{p-1}), \dots, \frac{\Delta T (\hat{f}_{M-2}^{p-1} - \hat{f}_M^{p-1})}{4(M-1)}, \frac{\Delta T}{4M} \hat{f}_{M-1}^{p-1}, \frac{\Delta T}{4(M+1)} \hat{f}_M^{p-1} \right]^\top \\
&\quad \mathbf{R}: (M+2) \times (M+2) \quad \mathbf{S}: (M+2) \times (M+1) \\
&= \begin{bmatrix} u_{T_n} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix} + \frac{\Delta T}{4} \begin{bmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & \frac{1}{2} & & & & \\ & & & \ddots & & & \\ & & & & \frac{1}{M-1} & & \\ & & & & & \frac{1}{M} & \\ & & & & & & \frac{1}{M+1} \end{bmatrix} \begin{bmatrix} 2 & -\frac{1}{2} & s_2 & s_3 & \cdots & s_M \\ 2 & 0 & -1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & 1 & 0 & -1 \\ 0 & 0 & 0 & \ddots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{f}_0^{p-1} \\ \hat{f}_1^{p-1} \\ \hat{f}_2^{p-1} \\ \hat{f}_3^{p-1} \\ \vdots \\ \hat{f}_M^{p-1} \end{bmatrix} \\
&= \mathbf{U}_0 + \frac{\Delta T}{4} \mathbf{R} \mathbf{S} \hat{\mathbf{f}}^p,
\end{aligned} \tag{4.5}$$

where \mathbf{R} and \mathbf{S} are the coefficient matrices and the first line of \mathbf{S} satisfies

$$s_m = (-1)^m \left(\frac{1}{m+1} - \frac{1}{m-1} \right), \quad m = 2, 3, \dots, M.$$

\mathbf{U}_0 and $\hat{\mathbf{f}}^p$ can be defined by

$$\begin{aligned}
\mathbf{U}_0 &= [u_{T_n}, 0, 0, \dots, 0]_{(M+2) \times 1}^\top, \\
\hat{\mathbf{f}}^p &= [\hat{f}_0^{p-1}, \hat{f}_1^{p-1}, \dots, \hat{f}_M^{p-1}]_{(M+1) \times 1}^\top.
\end{aligned} \tag{4.6}$$

For simplicity, we set the values of the function $f(t, u^{p-1}(t))$ on the Chebyshev-Gauss points by the notation $f_m^{p-1} = f(t_m, u_{p-1}^M(t_m))$, $m = 0, 1, \dots, M$, then (2.8) can be expressed as

$$\begin{aligned}
\hat{f}^{p-1} &= \begin{bmatrix} \frac{1}{M+1}(f_0^{p-1}T_0(\tau_0) + f_1^{p-1}T_0(\tau_1) + \dots + f_M^{p-1}T_0(\tau_M)) \\ \frac{2}{M+1}(f_0^{p-1}T_1(\tau_0) + f_1^{p-1}T_1(\tau_1) + \dots + f_M^{p-1}T_1(\tau_M)) \\ \vdots \\ \frac{2}{M+1}(f_0^{p-1}T_M(\tau_0) + f_1^{p-1}T_M(\tau_1) + \dots + f_M^{p-1}T_M(\tau_M)) \end{bmatrix} \\
&= \begin{matrix} \mathbf{V}:(M+1) \times (M+1) & \mathbf{T}_2:(M+1) \times (M+1) \end{matrix} \begin{bmatrix} \frac{1}{M+1} & & & \\ & \frac{2}{M+1} & & \\ & & \ddots & \\ & & & \frac{2}{M+1} \end{bmatrix} \begin{bmatrix} T_0(\tau_0) & T_0(\tau_1) & \dots & T_0(\tau_M) \\ T_1(\tau_0) & T_1(\tau_1) & \dots & T_1(\tau_M) \\ \vdots & \vdots & \ddots & \vdots \\ T_M(\tau_0) & T_M(\tau_1) & \dots & T_M(\tau_M) \end{bmatrix} \begin{bmatrix} f_0^{p-1} \\ f_1^{p-1} \\ \vdots \\ f_M^{p-1} \end{bmatrix} \\
&= \mathbf{V}\mathbf{T}_2\mathbf{f}^{p-1}.
\end{aligned} \tag{4.7}$$

where \mathbf{V} and \mathbf{T}_2 are coefficient matrices and the vector \mathbf{f}^{p-1} is defined by

$$\mathbf{f}^{p-1} = [f(t_0, u_M^{p-1}(t_0)), f(t_1, u_M^{p-1}(t_1)), \dots, f(t_M, u_M^{p-1}(t_M))]_{(M+1) \times 1}^\top. \tag{4.8}$$

To this end, given the initial guess \mathbf{u}_0 , the p -th matrix-vector iteration version of the Chebyshev-Gauss spectral collocation method for solving (2.6) is as follows.

$$\begin{aligned}
\mathbf{C}_\alpha &= \frac{1}{4}\mathbf{R}\mathbf{S}\mathbf{V}\mathbf{T}_2, \\
\hat{\mathbf{u}}^p &= \mathbf{U}_0 - \Delta T\mathbf{C}_\alpha\mathbf{f}^{p-1}, \\
\mathbf{u}^p &= \mathbf{T}_1\hat{\mathbf{u}}^p.
\end{aligned} \tag{4.9}$$

We develop the matrix-vector form of the method in order to acquire the stability function on the one hand, and to considerably improve computing efficiency so that it may be employed in our numerical experiments on the other. The matrix-vector form of the method improves the efficiency of computation sufficiently. Following, we establish the linear stability function for the Chebyshev-Gauss spectral collocation method, for which we initially derive the special form of the method for solving the linear equation,

$$\begin{aligned}
\mathbf{u}^p &= \mathbf{T}_1\hat{\mathbf{u}}^p = \mathbf{T}_1(\mathbf{U}_0 - \Delta T\mathbf{C}_\alpha\mathbf{f}^{p-1}) \\
&= \mathbf{T}_1\mathbf{U}_0 - \lambda\Delta T\mathbf{T}_1\mathbf{C}_\alpha\mathbf{u}^{p-1}.
\end{aligned} \tag{4.10}$$

By assuming $\mathbf{u}^* = [u^*(t_0), u^*(t_1), \dots, u^*(t_M)]^\top$ be the convergence solution after a sufficient number of iteration, we may rewrite (4.10) for given $z := \lambda \Delta T$ as

$$\begin{aligned}\mathbf{u}^* &= \mathbf{T}_1 \mathbf{U}_0 - z \mathbf{T}_1 \mathbf{C}_\alpha \mathbf{u}^*, \\ &= (\mathbf{I}_1 + z \mathbf{T}_1 \mathbf{C}_\alpha)^{-1} \mathbf{T}_1 \mathbf{U}_0,\end{aligned}\tag{4.11}$$

where \mathbf{I}_1 is an identity matrix of size $(M+1)$, then we can derive the corresponding coefficient vector $\hat{\mathbf{u}}^* = [u_1^*, u_2^*, \dots, u_{M+1}^*]^\top$ directly by (4.9)

$$\hat{\mathbf{u}}^* = \mathbf{U}_0 - \Delta T \mathbf{C}_\alpha (\lambda \mathbf{u}^*) = \mathbf{U}_0 - z \mathbf{C}_\alpha (\mathbf{I}_1 + z \mathbf{T}_1 \mathbf{C}_\alpha)^{-1} \mathbf{T}_1 \mathbf{U}_0.\tag{4.12}$$

Finally, the stability function of Chebyshev-Gauss spectral collocation method $\mathcal{R}_{\text{CG}}(z, M)$ could be acquired by the compression relations

$$\mathcal{R}_{\text{CG}}(z, M) = \frac{u^*(T_{n+1})}{u_{T_n}} = \frac{\mathbf{T} \hat{\mathbf{u}}^*}{u_{T_n}} = \mathbf{T} (\mathbf{I}_2 - z \mathbf{C}_\alpha (\mathbf{I}_1 + z \mathbf{T}_1 \mathbf{C}_\alpha)^{-1} \mathbf{T}_1) \mathbf{E},\tag{4.13}$$

where \mathbf{I}_2 is an $(M+2)$ dimensional identity matrix, \mathbf{T} and \mathbf{E}_1 are two vectors defined by $\mathbf{T} = [1, \dots, 1, 1]_{1 \times (M+2)}$ and $\mathbf{E} = [1, 0, \dots, 0]_{(M+2) \times 1}^\top$. ■

We can obtain the stability functions \mathcal{R}_{CG} , particularly when $M = 0$ and $M = 1$.

$$\mathcal{R}_{\text{CG}}(z, 0) = \frac{2-z}{2+z}, \quad \mathcal{R}_{\text{CG}}(z, 1) = \frac{z^2 - 8z + 16}{z^2 + 8z + 16}.\tag{4.14}$$

For $M = 0, 1, 2, 4, 20$, we displayed $|\mathcal{R}_{\text{CG}}(z, M)|$ as the functions of $z_{\max} := \Delta T \lambda_{\max}$ in Figure 4.1, which shows that for various M we have

$$\lim_{z \rightarrow \infty} |\mathcal{R}_{\text{CG}}(z)| = 1.\tag{4.15}$$

As a result, we can obtain the contraction factor of Parareal-CG method and derive the convergence analysis.

4.2 Convergence analysis of Parareal-CG method

The following conclusion regarding the convergence of Parareal-CG Algorithm on sufficiently long time intervals can be deduced directly from the analysis provided by Gander and Vandewalle [17] for the linear system of ODEs (1.2) with $A \in \mathbb{R}^{m \times m}$ and symmetric positive definite matrix (which therefore can be diagonalized and all the eigenvalues are positive real numbers).

Theorem 4.1 ([17]) *Let \mathcal{F}_{CG} be the Chebyshev-Gauss spectral collocation propagator with the stability function $\mathcal{R}_{\text{CG}}(z)$ (4.13), and let $\sigma(A) = \{\lambda_1, \dots, \lambda_m\}$ be the set of eigenvalues of the matrix A in (1.2). Then, the errors $\{e_n^k\}$ of the parareal-CG algorithm at k -th iteration using the backward-Euler method as the coarse*

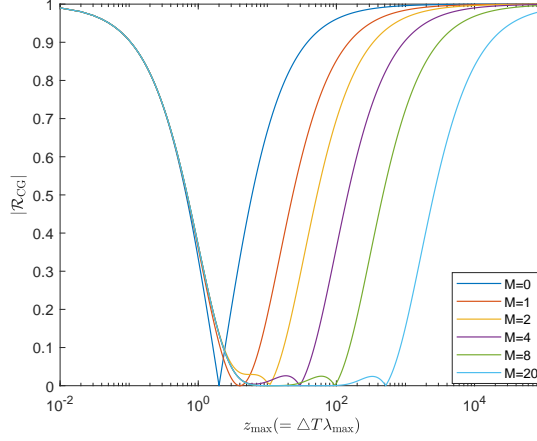


Figure 4.1: The contraction factor $|\mathcal{R}_{CG}|$ as a function of z_{\max} , for different M .

propagator satisfy

$$\sup_n \|V e_n^k\|_\infty \leq \rho^k \sup_n \|V e_0^k\|_\infty, \quad \rho = \max_{\lambda \in \sigma(A)} \mathcal{K}(\Delta T \lambda, M), \quad (4.16)$$

where ρ is the convergence factor of the parareal algorithm, $k \geq 1$ is the iteration index, and $V \in \mathbb{R}^{m \times m}$ consists of the eigenvectors of A (i.e., $V^{-1}AV = \text{diag}(\lambda_1, \dots, \lambda_m)$). The argument \mathcal{K}_{CG} , which is the convergence factor corresponding to a single eigenvalue (or in short "contraction factor" hereafter), is defined by

$$\mathcal{K}_{CG}(z, J) = \frac{\left| \mathcal{R}_{CG}(z, M) - \frac{1}{1+z} \right|}{1 - \left| \frac{1}{1+z} \right|}. \quad (4.17)$$

The convergence theorem 4.1 states that the behavior of the Parareal-CG algorithm over a long time interval is determined by the convergence factor $\rho(M)$ defined by $\rho(M) := \max_{z \in [0, z_{\max}]} \mathcal{K}(z, M)$ where $z_{\max} = \lambda_{\max} \Delta T$ and the quantity λ_{\max} denotes the maximal eigenvalue (or an upper bound) of the coefficient matrix A in (1.2). We can infer from the equation (4.16) that the smaller $\rho(M)$ is, the faster the algorithm converges. To maintain the convergence rate, the convergence factor $\rho(M)$ prefers to be around $\frac{1}{3}$.

The convergence factor of Parareal-CG algorithm $\mathcal{K}_{CG}(z, M)$ as functions of $z_{\max} := \Delta T \lambda_{\max}$ for various M are shown in Figure 4.2. We may infer from the behavior of the $\mathcal{K}_{CG}(z, M)$ which is similar to the convergence factor of Parareal-TR and Parareal-Gauss 4 algorithms discussed in [42] that

$$\lim_{z \rightarrow 0} \mathcal{K}_{CG}(z) = 0, \quad \lim_{z \rightarrow \infty} \mathcal{K}_{CG}(z) = 1. \quad (4.18)$$

It implies that the Parareal-CG algorithm cannot keep the convergence rate for arbitrarily large z , however, by solving an appropriately designed M that is large enough, it is still possible to get a feasible parareal solver. In further detail, as M grows, the convergence factor $\rho(M) := \max_{z \in [0, z_{\max}]} \mathcal{K}_{CG}(z, M)$ tends to be around $\frac{1}{3}$ for some given z .

The Convergence analysis can be derived by the convergence theorem 4.1 of Parareal-CG algorithm and the stability function (4.1) of Chebyshev-Gauss spectral collocation method.

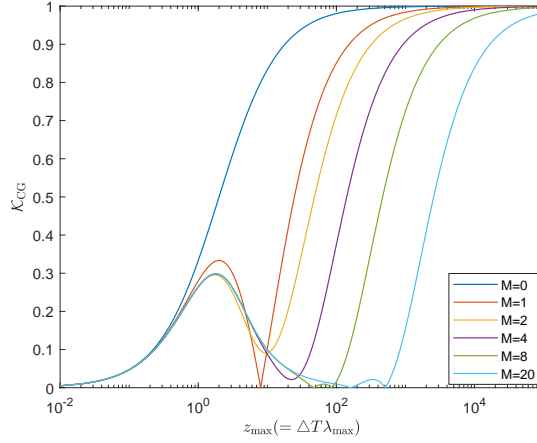


Figure 4.2: The contraction factor \mathcal{K}_{CG} as a function of z_{\max} , for different M .

Theorem 4.2 *Given a fixed constant $z_{\max} = \Delta T \lambda_{\max} > 0$, there exists some positive integer M_{\min}^* such that.*

$$\max_{z \in [0, z_{\max}]} \mathcal{K}_{CG}(z, M) \leq \frac{1}{3}, \quad \text{if } M \geq M_{\min}^*, \quad (4.19)$$

where \mathcal{K}_{CG} is the contraction factors of the Parareal-CG algorithm defined by (4.17) with the stability functions of Chebyshev-Gauss spectral collocation method \mathcal{R}_{CG} given in (4.13).

The Parareal-CG algorithm's lower bound, M_{\min}^* , is provided by

$$M_{\min}^* = \begin{cases} 0 & \text{if } z \leq z_0^*, \\ 1 & \text{if } z_0^* < z \leq z_1^*, \\ M_{CG}^* & \text{otherwise.} \end{cases} \quad (4.20)$$

where $M_{CG}^* > 1$ depends on z_{\max} and is the minimum positive integer which satisfies the following equation

$$|\mathcal{R}_{CG}(z_{\max}, M)| \leq \frac{3 + z_{\max}}{3(1 + z_{\max})}. \quad (4.21)$$

Moreover, the quantity z_0^* is the unique positive root of $\mathcal{K}_{CG}(z, 0) = \frac{1}{3}$ and z_1^* is the maximum positive root of $\mathcal{K}_{CG}(z, 1) = \frac{1}{3}$, that is to say, $z_0^* = 1$ and $z_1^* = 8 + 6\sqrt{2}$.

Proof. When $M = 0$, we can derive the $\mathcal{R}_{CG}(z, 0)$ and $\mathcal{K}_{CG}(z, 0)$ by

$$\mathcal{R}_{CG} = \frac{2 - z}{2 + z}, \quad \mathcal{K}_{CG}(z, 0) = \frac{z}{2 + z}, \quad (z > 0). \quad (4.22)$$

It is obvious that $\mathcal{K}_{CG}(z, 0)$ is an increasing function with respect to $z > 0$ and $\mathcal{K}_{CG}(z, 0) = \frac{1}{3}$ has the unique root $z_0^* = 1$. that is to say,

$$\max_{z \in [0, z_{\max}]} \mathcal{K}_{CG}(z, 0) \leq \begin{cases} \frac{1}{3}, & z \in (0, z_0^*] \\ \frac{z_{\max}}{2 + z_{\max}}, & z \in (z_0^*, z_{\max}]. \end{cases} \quad (4.23)$$

When $M = 1$, we can derive the $\mathcal{R}_{CG}(z, 1)$ and $\mathcal{K}_{CG}(z, 1)$ by

$$\mathcal{R}_{CG} = \frac{(4 - z)^2}{(4 + z)^2}, \quad \mathcal{K}_{CG}(z, 1) = \frac{|z^2 - 8z|}{(4 + z)^2}, \quad z > 0. \quad (4.24)$$

There are three roots $z_{11}^* = z_{12}^* = 2$ and $z_1^* = 8 + 6\sqrt{2}$ of $\mathcal{K}_{CG}(z, 1) = \frac{1}{3}$, respectively. $z_{11}^* = z_{12}^* = 2$ is also the maximum value point of $\mathcal{K}_{CG}(z, 1)$. Moreover, $\mathcal{K}_{CG}(z, 1)$ is an increasing function for $z \geq 8 + 6\sqrt{2}$. In conclusion, we may obtain

$$\max_{z \in [0, z_{\max}]} \mathcal{K}_{CG}(z, 0) \leq \begin{cases} \frac{1}{3}, & z \in (0, z_1^*] \\ \frac{z_{\max}^2 - 8z_{\max}}{(4 + z_{\max})^2}, & z \in (z_1^*, z_{\max}]. \end{cases} \quad (4.25)$$

When $M \geq 2$, we can infer from Figure 4.2 that there exists only one root depends on M $z^*(M) > z_1^*$ of $\mathcal{K}_{CG}(z, M) = \frac{1}{3}$ and $\mathcal{K}_{CG}(z, M)$ is an increasing function for $z \geq z^*(M)$, then we have

$$\max_{z \in [0, z_{\max}]} \mathcal{K}_{CG}(z, M) = \begin{cases} \frac{1}{3}, & z \in (0, z^*(M)] \\ \mathcal{K}_{CG}(z_{\max}, M) & z \in (z^*(M), z_{\max}]. \end{cases} \quad (4.26)$$

Therefore, we can derive that Theorem 4.2 holds. ■

We can infer from Theorem 4.2 that M has a significant impact on the convergence rates because the lower bound M_{\min}^* increases as $z_{\max} = \Delta T \lambda_{\max}$ grows. Using *while* loops in *Matlab*, we can derive M_{\min}^* for a given z_{\max} since M_{\min}^* is a natural number.

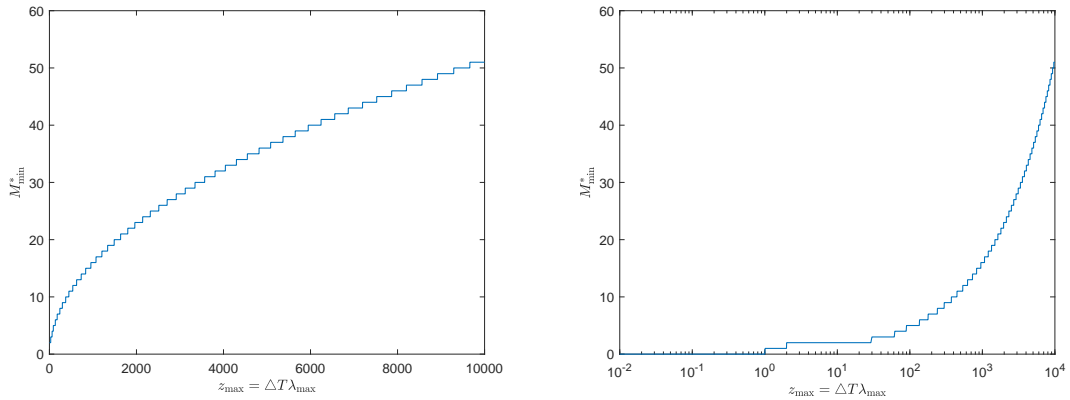


Figure 4.3: M_{\min}^*

Remark 4.1 (Comparison to other Parareal algorithms) *Researchers have proved that the following proposition for the Parareal-Euler [30] and Parareal-TR/BDF2 [35] algorithms utilizing the backward Euler and TR/BDF2 (i.e. ode23 solver for ODEs in MATLAB), respectively, as the \mathcal{F} propagators in the earlier papers.*

$$\max_{z \in [0, z_{\max}]} \mathcal{K}_{\text{Euler, TR/BDF2}}(z, J) < \frac{1}{3}, \quad \forall z_{\max} > 0, \quad (4.27)$$

which implies that for all $z_{\max} > 0$, the convergence rates of Parareal algorithms $\rho(J) < \frac{1}{3}$.

Unfortunately, not all methods yield such a uniform result. The following conclusion holds for the Parareal-TR and Parareal-Gauss4 algorithms [42] when utilizing the trapezoidal rule and fourth-order Gauss Runge-Kutta method as \mathcal{F} -propagator.

$$\max_{z \in [0, z_{\max}]} \mathcal{K}_{\text{TR, Gauss4}}(z, J) < \frac{1}{3}, \quad \text{if } J \text{ is even and } J \geq J_{\min}^*, \quad (4.28)$$

It indicates that if the mesh ratio J is large enough, the two parareal algorithms will converge as fast as the Parareal-Euler method. Our Parareal-CG method behaves similarly as Parareal-TR and Parareal-Gauss4 algorithms.

For the explicit \mathcal{F} -propagator forward Euler and fourth-order explicit Runge-Kutta algorithms used in the Parareal-fEuler and Parareal-4ERK algorithms, respectively, the convergence factor \mathcal{K} begins to suddenly trend to infinity at some z_{\max} . Therefore, these parareal algorithms can be used to solve a very limited number of equations.

$$\mathcal{K}_{\text{fEuler, 4ERK}}(z, J) = \infty, \quad z > z^*. \quad (4.29)$$

The convergence factors of the three kinds of parareal algorithms are shown in Figure 4.4,

5 Numerical Experiment

In this section we verify the convergence and efficiency of the Parareal-CG algorithm. In all the computations, the initial iteration for the parareal algorithm is chosen randomly and the iteration process stops when the following tolerance is obtained:

$$\max_n \|u_n^{k+1} - u_n^k\|_{\infty} \leq 10^{-10}. \quad (5.1)$$

Moreover, the absolute error and iteration error in all experiments are defined by

$$\text{Absolute error : } \max_n \|u_n^{k+1} - u(T_n)\|_{\infty}. \quad \text{Iteration error : } \max_n \|u_n^{k+1} - u_n^k\|_{\infty}. \quad (5.2)$$

Example 5.1 (Near Earth satellite motion integration problem) *We take a two-body motion in low Earth orbit (LEO) with just mutual gravitational attraction for the first example. The system of the second*

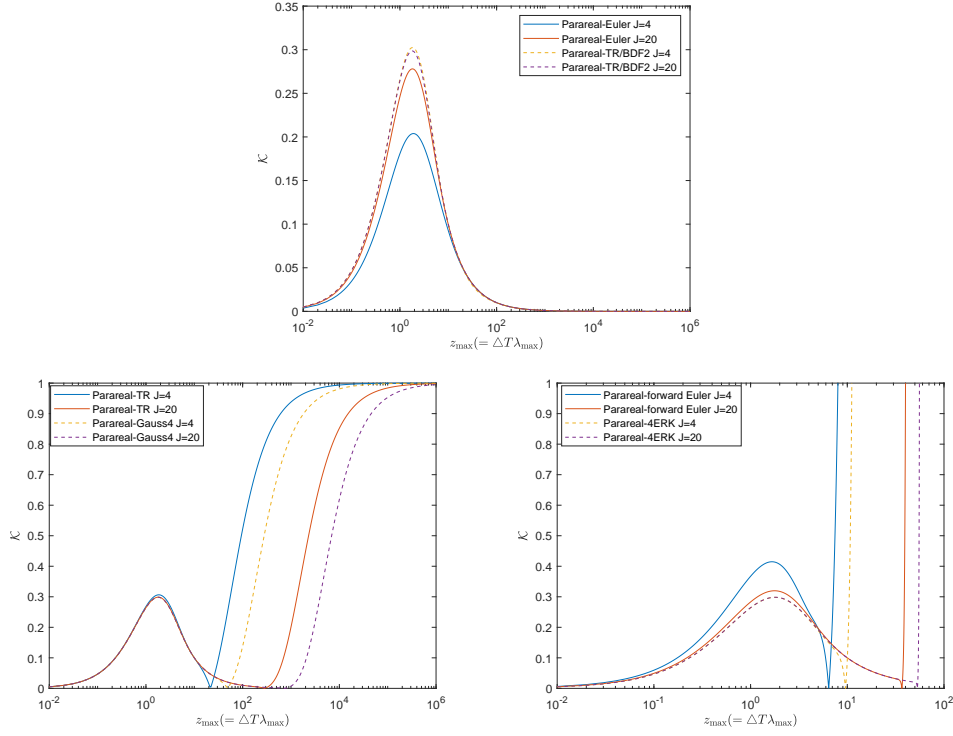


Figure 4.4: The contraction factors \mathcal{K} as a functions of z_{\max} , for different M . Top: Parareal-Euler (\mathcal{F} =backward Euler) and Parareal-TR/BDF2 (\mathcal{F} =TR/BDF2). Bottom left side: Parareal-TR (\mathcal{F} =trapezoidal rule) and Parareal-Gauss4 (\mathcal{F} =forth-order Gauss). Bottom right side: Parareal-fEuler (\mathcal{F} =forward Euler) and Parareal-4ERK (\mathcal{F} =forth-order explicit Runge-Kutta).

order three-dimensional equations is

$$\begin{cases} x''(t) = -\frac{\mu}{r(t, x, y, z)^3}x(t), & t \in [0, 50], \\ y''(t) = -\frac{\mu}{r(t, x, y, z)^3}y(t), & t \in [0, 50], \\ z''(t) = -\frac{\mu}{r(t, x, y, z)^3}z(t), & t \in [0, 50], \end{cases} \quad (5.3)$$

where x , y and z are the three coordinates in some Earth-centered inertial reference frame; r is the distance between the two bodies defined by

$$r(t, x, y, z) = \sqrt{x(t)^2 + y(t)^2 + z(t)^2};$$

$\mu = 3.986 \times 10^5 \text{ km}^3/\text{s}^2$ is the Earth gravitational constant. We formulate the equations as a first-order system of six-dimensional differential equations, We formulate the equations as a first-order system of six-dimensional differential equations, where the solution is uniquely determined by the initial position and

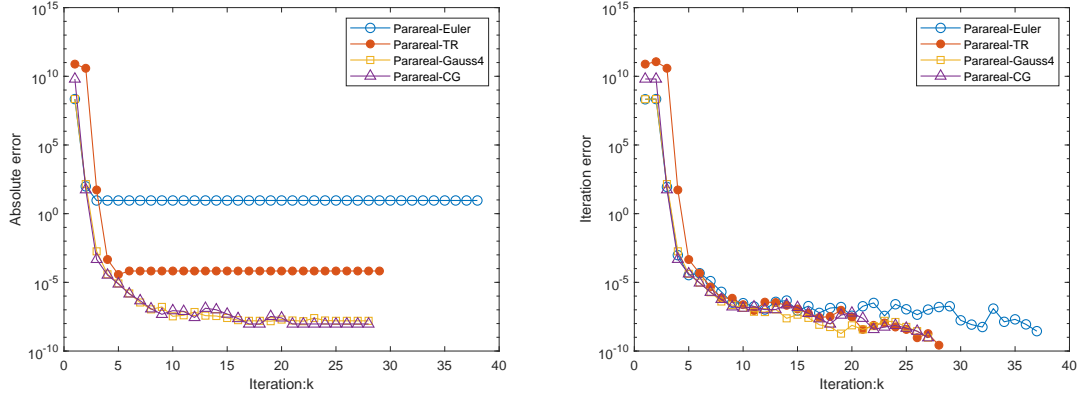


Figure 5.5: Comparisons of the absolute errors (left) and iteration errors (right) for the Parareal-Euler, Parareal-TR and Parareal-Gauss4 and Parareal-CG algorithms.

velocity

$$\begin{aligned} [x(0), y(0), z(0)] &= [464.856, 6667.880, 574.231]km, \\ [x'(0), y'(0), z'(0)] &= [-2.8381188, -0.7871898, 7.0830275]km/s. \end{aligned}$$

We obtain the exact solution of the equation (5.3) by solving the two-body Keplerian motion using the F and G approach in [33].

We compare the accuracy of Parareal-CG, Parareal-Euler, Parareal-TR/BDF2 and Parareal-Gauss4 algorithms revisited in Remark 4.1 for solving the example. In each coarse grid, we fix $\Delta T = 0.25$ with $M = 6$ for Parareal-CG algorithm and $J = 6$ for other parareal algorithms. Then we perform the absolute errors and iteration errors of the parareal algorithms in Figure 5.5. In this way, we can draw the following conclusions by the convergence behaviors in the figures.

- In comparison to the Parareal-Euler and Parareal-TR, the Parareal-CG algorithm utilizes less iterations. While the Parareal-Gauss4 algorithm uses the same number of iterations as Parareal-CG algorithm.
- Using the same number of points $M = J = 6$ in every coarse grid, the Parareal-CG algorithm has a higher accuracy than the three others.

Example 5.2 (Burgers' equation) Consider the 1D Burgers' equation with initial and boundary condition as following

$$\begin{cases} \frac{\partial u}{\partial t} - \nu \frac{\partial^2 u}{\partial x^2} + u \frac{\partial u}{\partial x} = 0, & (x, t) \in [0, 2) \times [0, 4], \\ u(x, 0) = \frac{2\nu\pi \sin(\pi x)}{\alpha + \cos(\pi x)}, & x \in [0, 2), \\ u(0, t) = u(2, t) = 0, & t \in [0, 4], \end{cases} \quad (5.4)$$

in our computations, we choose $\alpha = 2$ and test $\nu = 0.05, 0.005$. The exact solution of the equation is

$$u(x, t) = \frac{2\nu\pi \exp(-\pi^2\nu t) \sin(\pi x)}{\alpha + \exp(-\pi^2\nu t) \cos(\pi x)}. \quad (5.5)$$

We divide the spatial domain $x \in [0, 2)$ into an N_x mesh uniformly with $\Delta x = \frac{2}{N_x}$, in this way, we have $x_j = j\Delta x$, $j = 0, 1, \dots, N_x - 1$. Applying the fourth-order compact finite difference scheme to approximate $\frac{\partial u}{\partial x}$ and $\frac{\partial^2 u}{\partial x^2}$, we have

$$\begin{aligned} \frac{1}{6} \frac{\partial u}{\partial x}(x_{j+2}, t) + \frac{2}{3} \frac{\partial u}{\partial x}(x_{j+1}, t) + \frac{1}{6} \frac{\partial u}{\partial x}(x_j, t) &= \frac{1}{2\Delta x} (u(x_{j+2}, t) - u(x_j, t)), \\ \frac{1}{12} \frac{\partial^2 u}{\partial x^2}(x_{j+2}, t) + \frac{5}{6} \frac{\partial^2 u}{\partial x^2}(x_{j+1}, t) + \frac{1}{12} \frac{\partial^2 u}{\partial x^2}(x_j, t) &= \frac{1}{\Delta x^2} (u(x_{j+2}, t) - 2u(x_{j+1}, t) + u(x_j, t)). \end{aligned} \quad (5.6)$$

Define the vectors

$$\begin{aligned} \dot{\mathbf{u}} &= \left[\frac{\partial u(x_0, t)}{\partial t}, \frac{\partial u(x_1, t)}{\partial t}, \dots, \frac{\partial u(x_{N_x-1}, t)}{\partial t} \right]^\top, \\ \mathbf{u} &= [u(x_0, t), u(x_1, t), \dots, u(x_{N_x-1}, t)]^\top. \end{aligned} \quad (5.7)$$

Then combining the compact finite difference scheme with the period boundary condition, the Burgers' equation (5.4) has the following spatial semidiscrete scheme

$$\dot{\mathbf{u}} + A_1 \mathbf{u} + \mathbf{u} \cdot (A_2 \mathbf{u}) = 0, \quad (5.8)$$

where the coefficient matrices A_1 and A_2 are

$$\begin{aligned} A_1 &= -\frac{\nu}{\Delta x^2} \begin{bmatrix} \frac{5}{6} & \frac{1}{12} & & \frac{1}{12} \\ \frac{1}{12} & \frac{5}{6} & \frac{1}{12} & \\ & \ddots & \ddots & \ddots \\ & & \frac{1}{12} & \frac{5}{6} & \frac{1}{12} \\ \frac{1}{12} & & \frac{1}{12} & \frac{5}{6} \end{bmatrix}^{-1} \begin{bmatrix} -2 & 1 & & 1 \\ 1 & -2 & 1 & \\ & \ddots & \ddots & \ddots \\ & & 1 & -2 & 1 \\ 1 & & & 1 & -2 \end{bmatrix}, \\ A_2 &= \frac{1}{2\Delta x} \begin{bmatrix} \frac{2}{3} & \frac{1}{6} & & \frac{1}{6} \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & \\ & \ddots & \ddots & \ddots \\ & & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \frac{1}{6} & & \frac{1}{6} & \frac{2}{3} \end{bmatrix}^{-1} \begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & \ddots \\ & & -1 & 0 & 1 \\ 1 & & & -1 & 0 \end{bmatrix}. \end{aligned}$$

The three figures in Fig 5.6 show the dependence of the convergence rate of the Parareal-CG algorithm on ΔT , Δx , M , respectively. We can draw the following conclusions by the figure.

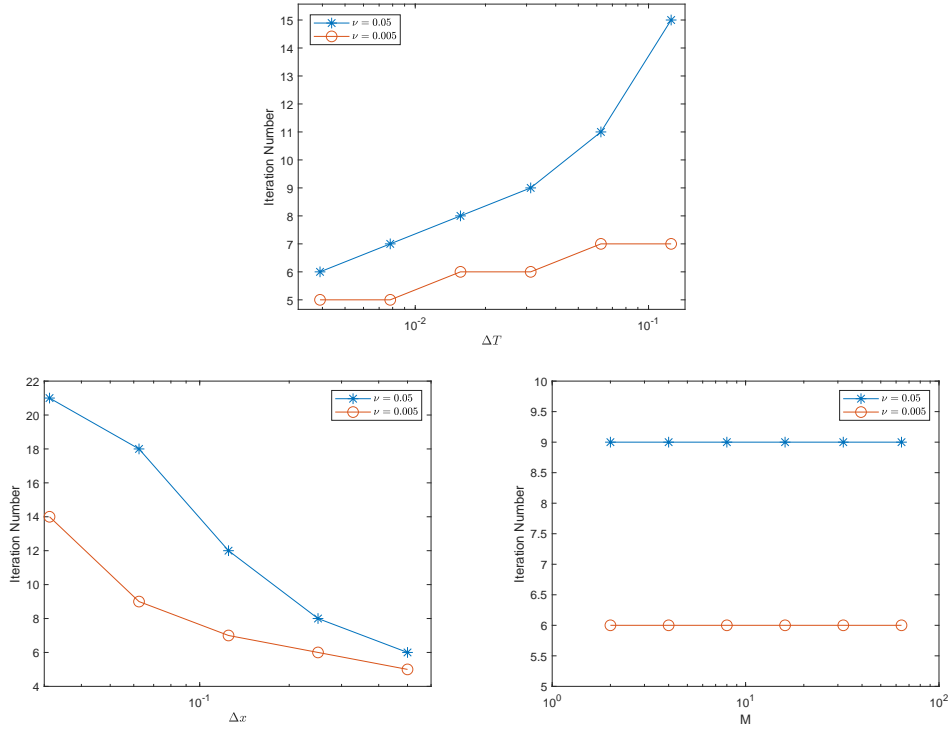


Figure 5.6: Top: The semi-log plot of the dependency of the Parareal-CG algorithm's convergence rate on ΔT , $\Delta x = 1/4$, $M = 4$, ΔT ranges from 2^{-3} to 2^{-8} . Bottom left side: The semi-log plot of the dependency of the Parareal-CG algorithm's convergence rate on Δx , $\Delta T = 1/64$, $M = 4$, Δx ranges from 2^{-1} to 2^{-5} . Bottom right side: The semi-log plot of the dependency of the Parareal-CG algorithm's convergence rate on M , $\Delta T = 1/32$, $\Delta x = 1/4$, M ranges from 2^1 to 2^6 .

- Each of the ν possesses robust convergence rate with respect to the change of ΔT . The convergence factor decreases as ΔT increases.
- Each of the ν possesses robust convergence rate with respect to the change of Δx . The convergence factor increases as Δx varies from small to large since the eigenvalues of A_1 and A_2 increases as Δx reduces.
- The convergence rate is insensitive to the choice of M with $\Delta T = 1/32$, which implies the high accuracy of the Chebyshev-Gauss spectral collocation method.
- All the experiments $\nu = 0.005$ needs less iteration number than $\nu = 0.05$.

6 Conclusion

In this paper, we propose the Parareal-CG method for solving time-dependent differential equations, where the coarse propagator \mathcal{G} is fixed by backward Euler method and the fine propagator \mathcal{F} is chosen to be Chebyshev-Gauss spectral collocation method. The algorithm does have a convergence factor around 0.333, although the number of Chebyshev-Gauss points M needs to be somewhat large. The spectral radius of the

matrix A and the ΔT provide the lower bound of M that guarantees such a expective convergence factor. Some numerical experiments illustrate the accuracy and convergency of the presented algorithm.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable suggestions, which helped us to improve this article greatly. This work was partially supported by the Postgraduate Scientific Research Innovation Project of Hunan Province (No.CX20210012).

References

- [1] X. Bai, *Modified Chebyshev-Picard iteration methods for solution of initial value and boundary value problems* (Ph. D. dissertation), Texas A&M University, College Station, TX, 2010.
- [2] L. Baffico, S. Bernard, Y. Maday, G. Turinici, and G. Zérah, *Parallel-in-time molecular-dynamics simulations*, Phys. Rev. E 66 (3) (2002) 057701.
- [3] X. Bai, J. L. Junkins, *Modified Chebyshev-Picard iteration methods for orbit propagation*, J. Astronaut. Sci. 58 (4) (2011) 583–613.
- [4] D. Q. Bui, C. Japhet, Y. Maday, and P. Omnes, *Coupling parareal with optimized Schwarz waveform relaxation for parabolic problems*, SIAM J. Numer. Anal. 60 (3) (2022) 913–939.
- [5] C.-E. Brehier, X. Wang, *On parareal algorithms for semilinear parabolic stochastic PDEs*, SIAM J. Numer. Anal. 58 (1) (2020) 254–278.
- [6] J. D. Cole, *On a quasi-linear parabolic equation occurring in aerodynamics*, Quart. Appl. Math. 9 (1951) 225–236.
- [7] C. W. Clenshaw, H. J. Norton, *The solution of nonlinear ordinary differential equations in Chebyshev series*, Comput. J. 6 (1963/64) 88–92.
- [8] S. Costanzo, T. Sayadi, M. F. de Pando, P. J. Schmid, and P. Frey, *Parallel-in-time adjoint-based optimization-application to unsteady incompressible flows*, J. Comput. Phys. 471 (2022) 111664.
- [9] X. Dai, C. Bris, F. Legoll and Y. Maday, *Symmetric parareal algorithms for Hamiltonian systems*, ESAIM Math. Model. Numer. Anal. 47 (3) (2013) 717–742.
- [10] F. Danieli, B. S. Southworth and A. J. Wathen, *Space-time block preconditioning for incompressible flow*, SIAM J. Sci. Comput. 44 (1) (2022) A337–A363.
- [11] F. Danieli, A. Wathen, *All-at-once solution of linear wave equations*, Numer. Linear Algebra Appl. 28 (6) (2021) e2386.
- [12] T. Feagin, P. Nacozy, *Matrix formulation of the Picard method for parallel computation*, Celestial Mech. 29 (2) (1983) 107–115.

- [13] L. Fang, S. Vandewalle and J. Meyers, *A parallel-in-time multiple shooting algorithm for large-scale PDE-constrained optimal control problems*, J. Comput. Phys. 452 (2022) 110926.
- [14] M. J. Gander *50 years of time parallel time integration*, in Multiple Shooting and Time Domain Decomposition, T. Cararro, M. Geiger, S. Körkel and R. Rannacher, eds. Springer-Verlag, Heidelberg, 2015.
- [15] M. J. Gander, E. Hairer, *Analysis for parareal algorithms applied to Hamiltonian differential equations*, J. Comput. Appl. Math. 259 (2014) 2–13.
- [16] I. C. Garcia, I. Kulchytska-Ruchka and S. Schöps, *Parareal for index two differential algebraic equations*, Numer. Algorithms 91 (1) (2022) 389–412.
- [17] M. J. Gander, S. Vandewalle, *Analysis of the parareal time-parallel time-integration method*, SIAM J. Sci. Comput. 29 (2) (2007) 556–578.
- [18] B. Guo, Z. Wang, *Legendre-Gauss collocation methods for ordinary differential equations*, Adv. Comput. Math. 30 (3) (2009) 249–280.
- [19] B. Guo, Z. Wang, *A spectral collocation method for solving initial value problems of first order ordinary differential equations*, Discrete Contin. Dyn. Syst. Ser. B 14 (3) (2010) 1029–1054.
- [20] M. J. Gander, S. Wu, *A diagonalization-based parareal algorithm for dissipative and wave propagation problems*, SIAM J. Numer. Anal. 58 (5) (2020) 2981–3009.
- [21] E. Hairer, G. Wanner, *Solving ordinary differential equation II: stiff and differential-algebraic problems*, Springer-Verlag, Berlin, 1991.
- [22] J. Hong, X. Wang and L. Zhang, *Parareal exponential θ -scheme for longtime simulation of stochastic Schrödinger equations with weak damping*, SIAM J. Sci. Comput. 41 (6) (2019) B1155–B1177.
- [23] Y. Jiang, J. Liu, *Fast parallel-in-time quasi-boundary value methods for backward heat conduction problems*, Appl. Numer. Math. 184 (2023) 325–339.
- [24] M. La Scala, A. Bose, *Relaxation/Newton methods for concurrent time step solution of differential-algebraic equations in power system dynamic simulations*, IEEE Trans. Circuits Systems I Fund. Theory Appl. 40 (5) (1993) 317–330.
- [25] C. Lohmann, J. Dünnebacke and S. Tured, *Fourier analysis of a time-simultaneous two-grid algorithm using a damped Jacobi waveform relaxation smoother for the one-dimensional heat equation*, J. Numer. Math. 30 (3) (2022), 173–207.
- [26] F. Legoll, T. Lelièvre, and U. Sharma, *An Adaptive Parareal Algorithm: Application to the Simulation of Molecular Dynamics Trajectories*, SIAM J. Sci. Comput. 44 (1) (2022) B146–B176.
- [27] J. Lions, Y. Maday and G. Turinici, *A "parareal" in time discretization of PDE's*, C. R. Acad. Sci. Paris Sér. I Math. 332 (7) (2001) 661–668.

- [28] J. Liu, Z. Wang, *A ROM-accelerated parallel-in-time preconditioner for solving all-at-once systems in unsteady convection-diffusion PDEs*, Appl. Math. Comput. 416 (2022) 126750.
- [29] J. Liu, X. Wang, S. Wu and T. Zhou, *A well-conditioned direct PinT algorithm for first- and second-order evolutionary equations*, Adv. Comput. Math. 48 (3) (2022) 1–29.
- [30] T. P. Mathew, M. Sarkis and C. E. Schaerer, *Analysis of block parareal preconditioners for parabolic optimal control problems*, SIAM J. Sci. Comput. 32 (3) (2010) 1180–1200.
- [31] K. Pentland, M. Tamborrino, D. Samaddar, and L. C. Appel, *Stochastic Parareal: An Application of Probabilistic Methods to Time-Parallelization*, SIAM J. Sci. Comput. (2022) S82–S102.
- [32] J. Shen, T. Tang and L. Wang, *Spectral Methods: algorithms, analysis and applications*, Springer, Heidelberg, 2011.
- [33] H. Schaub and J. L. Junkins, *Analytical Mechanics of Space Systems*, Reston, VA: American Institute of Aeronautics and Astronautics, Inc, 1 edition, 2003.
- [34] G. A. Staff, E. M. Rønquist, *Stability of the parareal algorithm*, in Domain Decomposition Methods in Science and Engineering, Lecture Notes Comput. Sci. Eng. 40 (2005) 449–456.
- [35] S. Wu, *Convergence analysis of some second-order parareal algorithms*, IMA J. Numer. Anal. 35 (3) (2015) 1315–1341.
- [36] S. Wu, T. Huang, *A fast second-order parareal solver for fractional optimal control problems*, J. Vib. Control 24 (15) (2018) 3418–3433.
- [37] R. Woollands, J. L. Junkins, *Nonlinear differential equation solvers via adaptive Picard-Chebyshev iteration: applications in astrodynamics*, J. Guid. Control Dyn. 42 (5) (2019) 1007–1022.
- [38] S. Wu, J. Liu, *A parallel-in-time block-circulant preconditioner for optimal control of wave equations*, SIAM J. Sci. Comput. 42 (3) (2020) A1510–A1540.
- [39] Z. Wang, J. Mu, *A multiple interval Chebyshev-Gauss-Lobatto collocation method for ordinary differential equations*, Numer. Math. Theor. Meth. Appl. 9 (4) (2016) 619–639.
- [40] R. Watschinger, M. Merta, G. Of and J. Zapletal, *A parallel fast multipole method for a space-time boundary element method for the heat equation*, SIAM J. Sci. Comput. 44 (4) (2022) C320–C345.
- [41] Y. Wang, G. Ni and Y. Liu, *Multistep Newton-Picard method for nonlinear differential equations*, J. Guid. Control Dyn. 43 (11) (2020) 2148–2155.
- [42] S. Wu, T. Zhou, *Convergence analysis for three parareal solvers*, SIAM J. Sci. Comput. 37 (2) (2015) A970–A992.
- [43] S. Wu, T. Zhou, *Parallel implementation for the two-stage SDIRK methods via diagonalization*, J. Comput. Phys. 428 (2021) 110076.

- [44] X. Yang, Z. Wang, *A Chebyshev-Gauss spectral collocation method for ordinary differential equations*, J. Comput. Math. 33 (1) (2015) 59–85.
- [45] J. Yang, Z. Yuan and Z. Zhou, *Robust convergence of parareal algorithms with arbitrarily high-order fine propagators*, SSRN Electronic Journal, 2022.
- [46] L. Zhang, J. Wang, W. Zhou, L. Liu and L. Zhang, *Convergence analysis of parareal algorithm based on Milstein scheme for stochastic differential equations*, J. Comput. Math. 38 (3) (2020) 487–501.